

FIG. 1

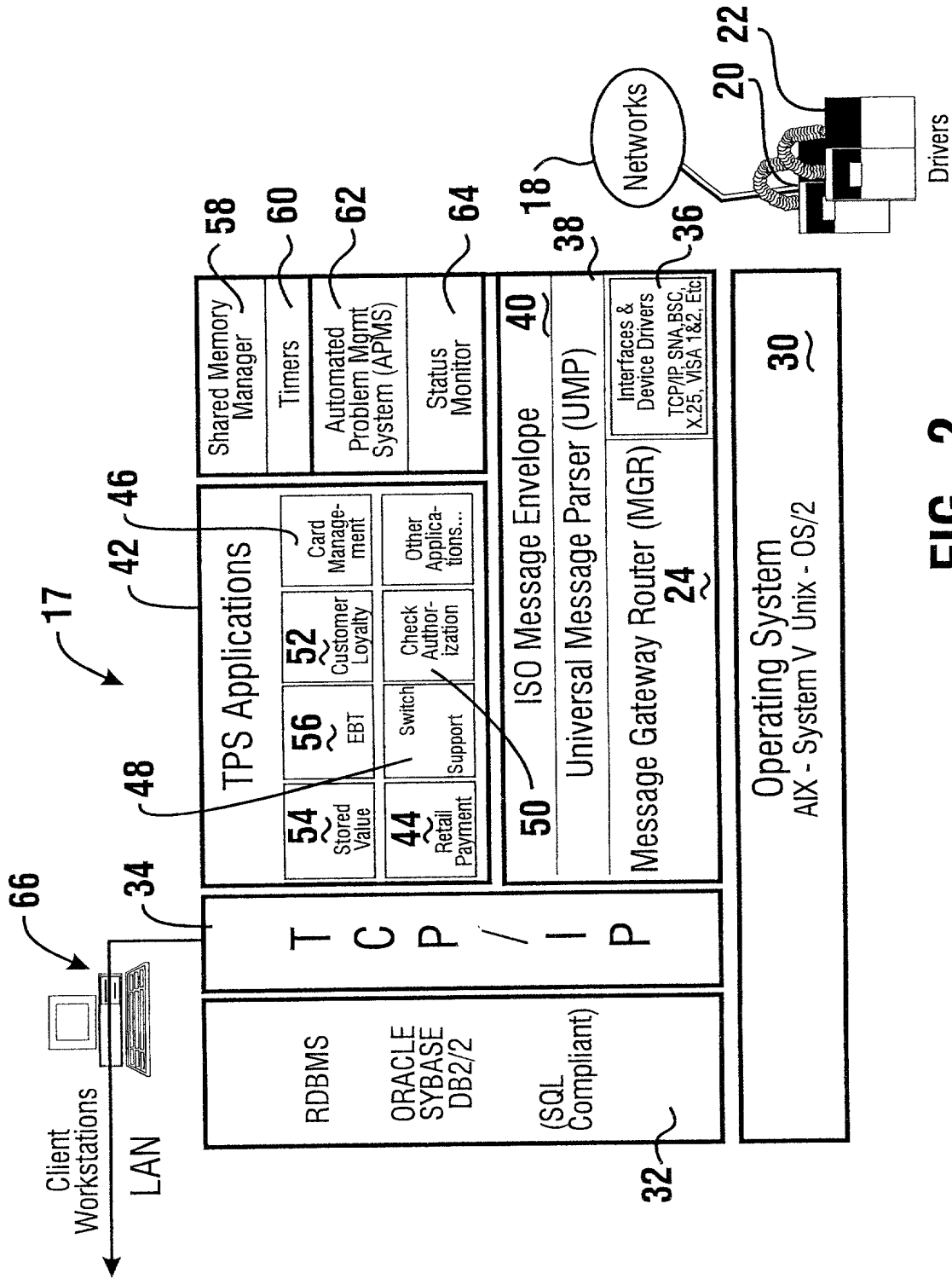


FIG. 2

106250 2372960

Standard Message Envelope (SME) Format.

1	Header Sid	Header Layout Version	1
2	Source Node Sid	The message originating node system Id.	6
3	Message Receive System Time	The System time in YYYYMMDDHHMISSmmm format.	17
4	Internal Message Sid	Unique system Id of the received message.	4
5	Service Sid	The Message Processing Program (MPP) service system Id, which can process received message.	4
6	Target Node Sid	The message receiving node system Id	6
7	Data Format Indicator (Source)	Message data format type 0 - External Data Source 1 - Internal Data Source	1
8	Message Direction	The direction of message routing.	1
9	Processing Time	Elapsed message processing time in milliseconds.	5
10	Processing Node Sid	The last processing node system Id	6
11	Target Line Node Sid	Line driver node system id. Assigned when terminal is attached to line group.	6
12	Message Text	The Message text in ISO8583 format	Variable

FIG. 3

FIG. 4

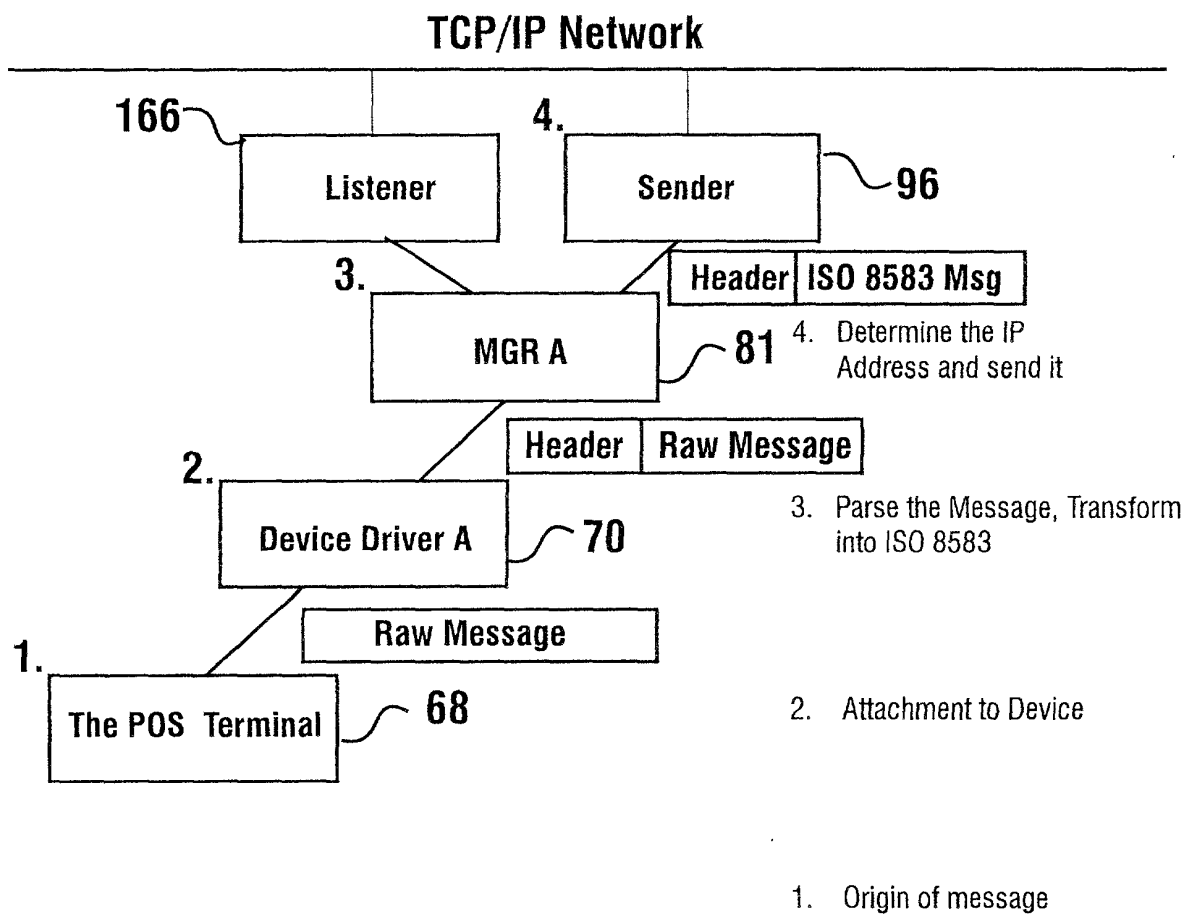
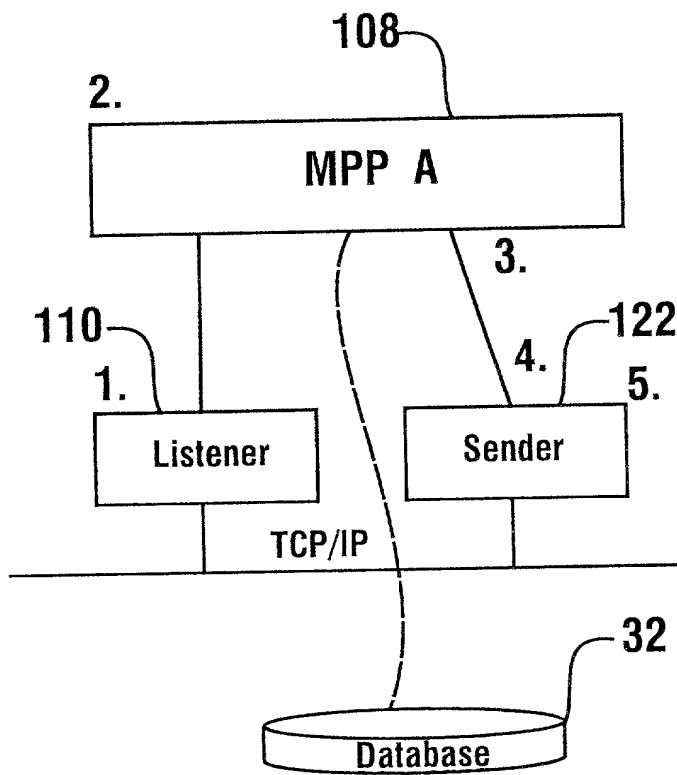
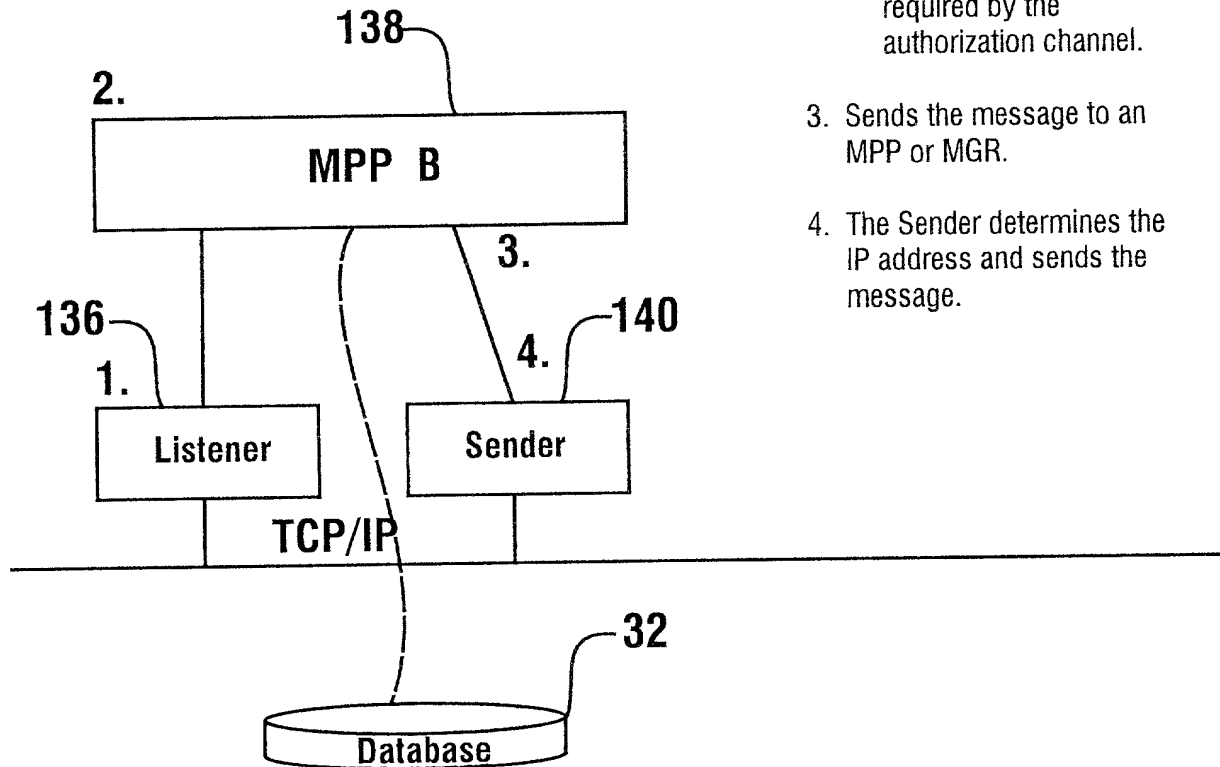


FIG. 4



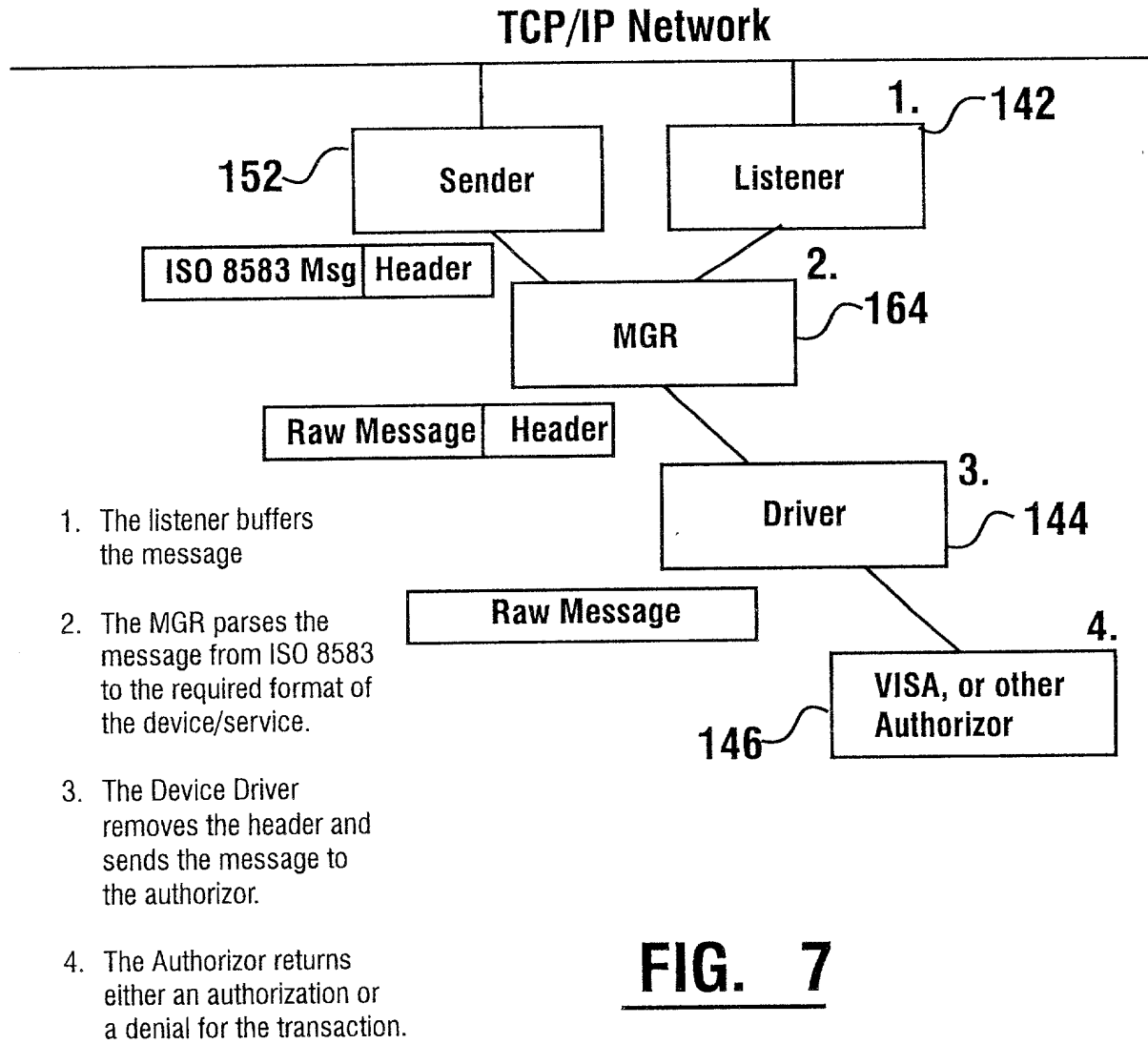
1. The Listener buffers the data, then places the data onto the input queue of the MPP.
2. The MPP performs various functions based upon the requirements of the message.
 - Builds an internal array.
 - Parses composite fields into subfields of the array.
 - May perform authorization.
 - Determines who to send the message to. May be an MPP or MGR
 - Builds a new message.
3. Sends a copy of the data to the database for archive.
4. Sends the message to the authorization host.
5. The Sender determines the IP address and sends the message.

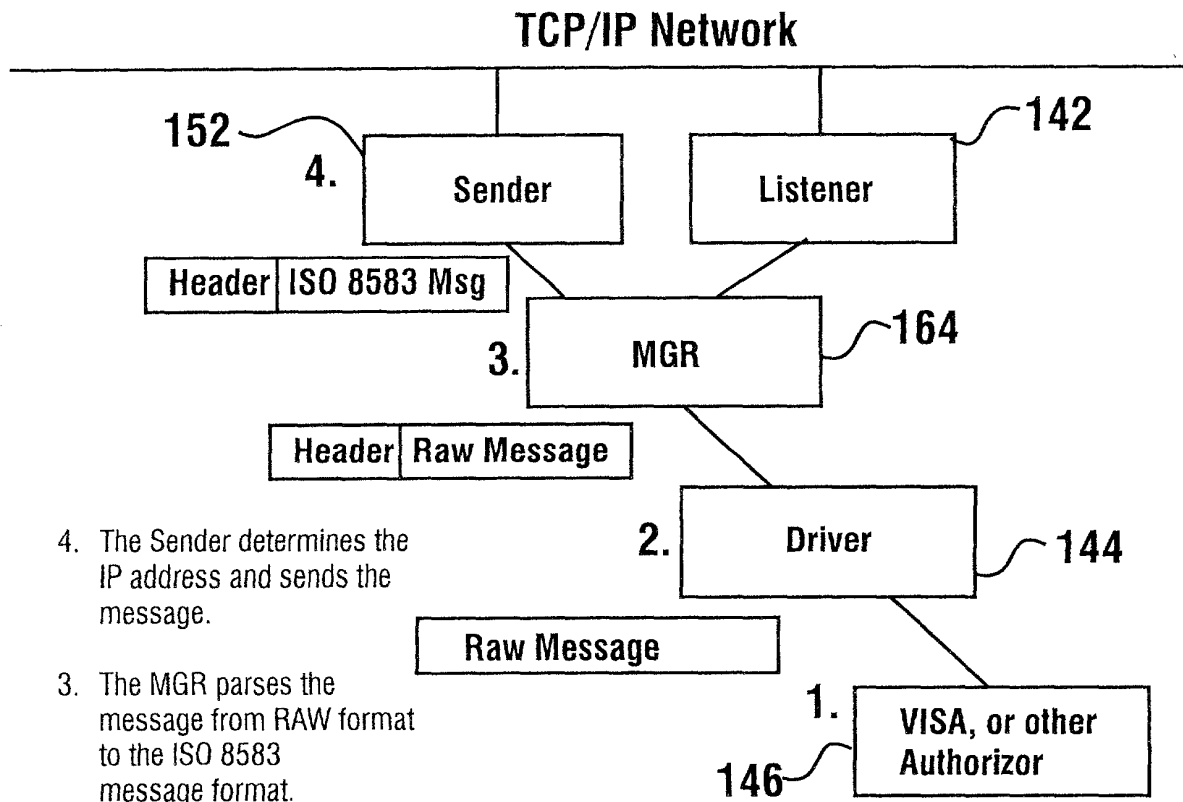
FIG. 5



1. The Listener buffers the data, then places the data onto the input queue of the MPP.
2. The MPP performs various functions based upon the requirements of the message.
 - Builds an internal array.
 - Builds any subfields required by the authorization channel.
3. Sends the message to an MPP or MGR.
4. The Sender determines the IP address and sends the message.

FIG. 6



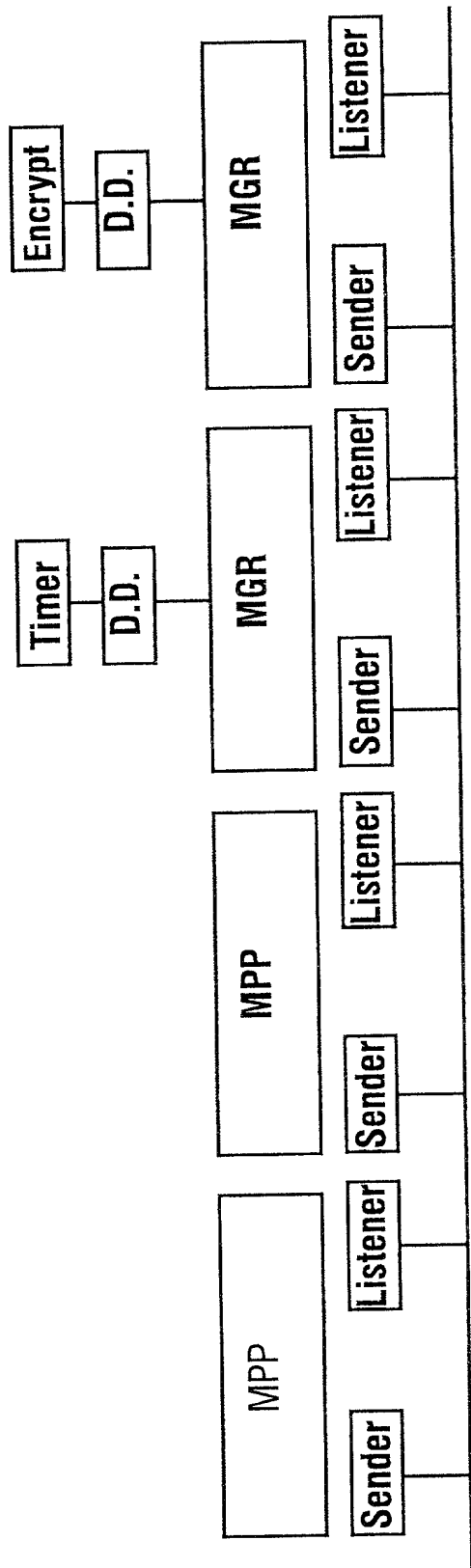


4. The Sender determines the IP address and sends the message.
3. The MGR parses the message from RAW format to the ISO 8583 message format.

2. The device driver add a header and fills some fields after getting the information directly from the network or host.

1. The Authorizer (VISA or other host) returns the message. This represents the actual host/network.

FIG. 8



1. The Message is sent to the second MPP. It uses a echo-back field to determine the origin of the message. The database contains the original message with a key. It may send the message to the first MPP by calling the Encryption Device for decryption of the PAN.
2. The message is received by the first MPP. It may need to build special fields, such as track II data. It will then send the message back to the original calling device by using the saved data in the database.

FIG. 9

1052337 2312960

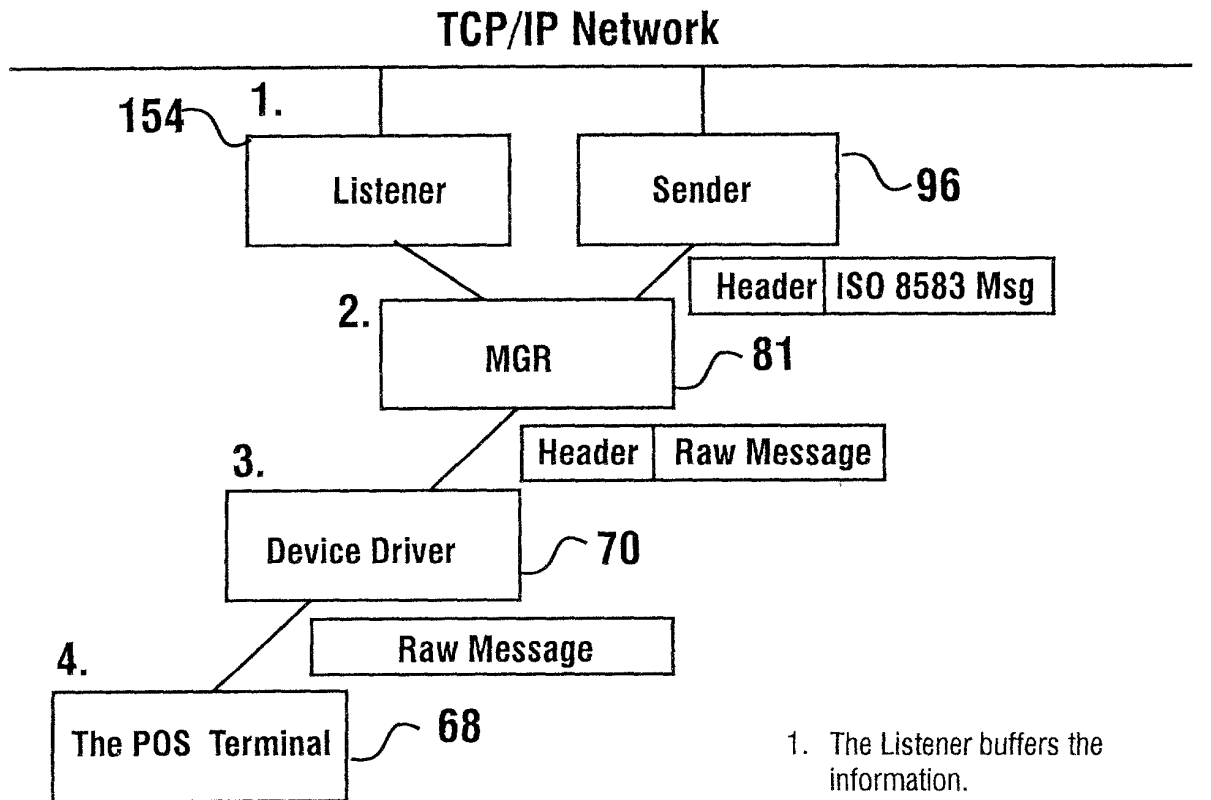
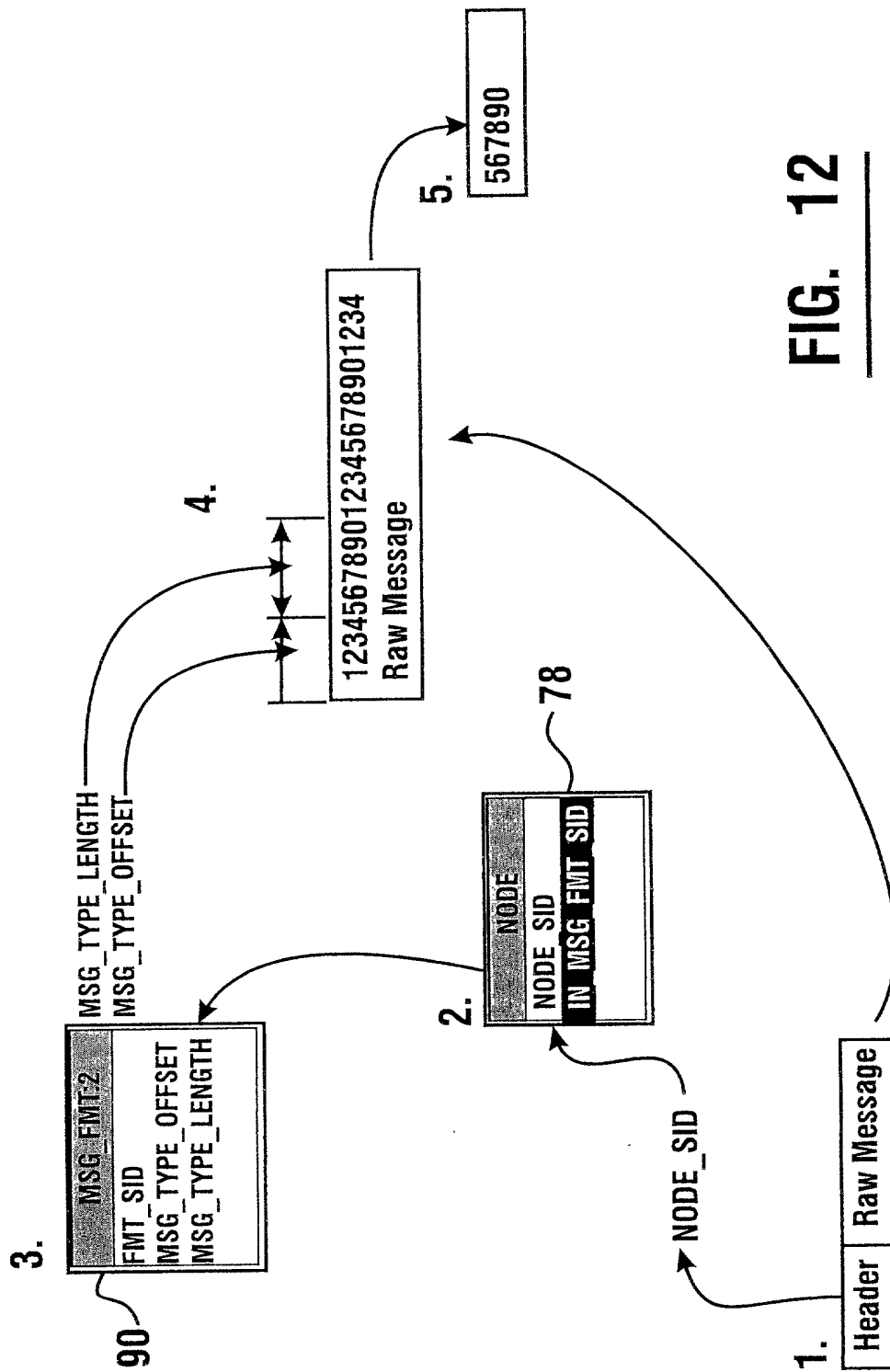
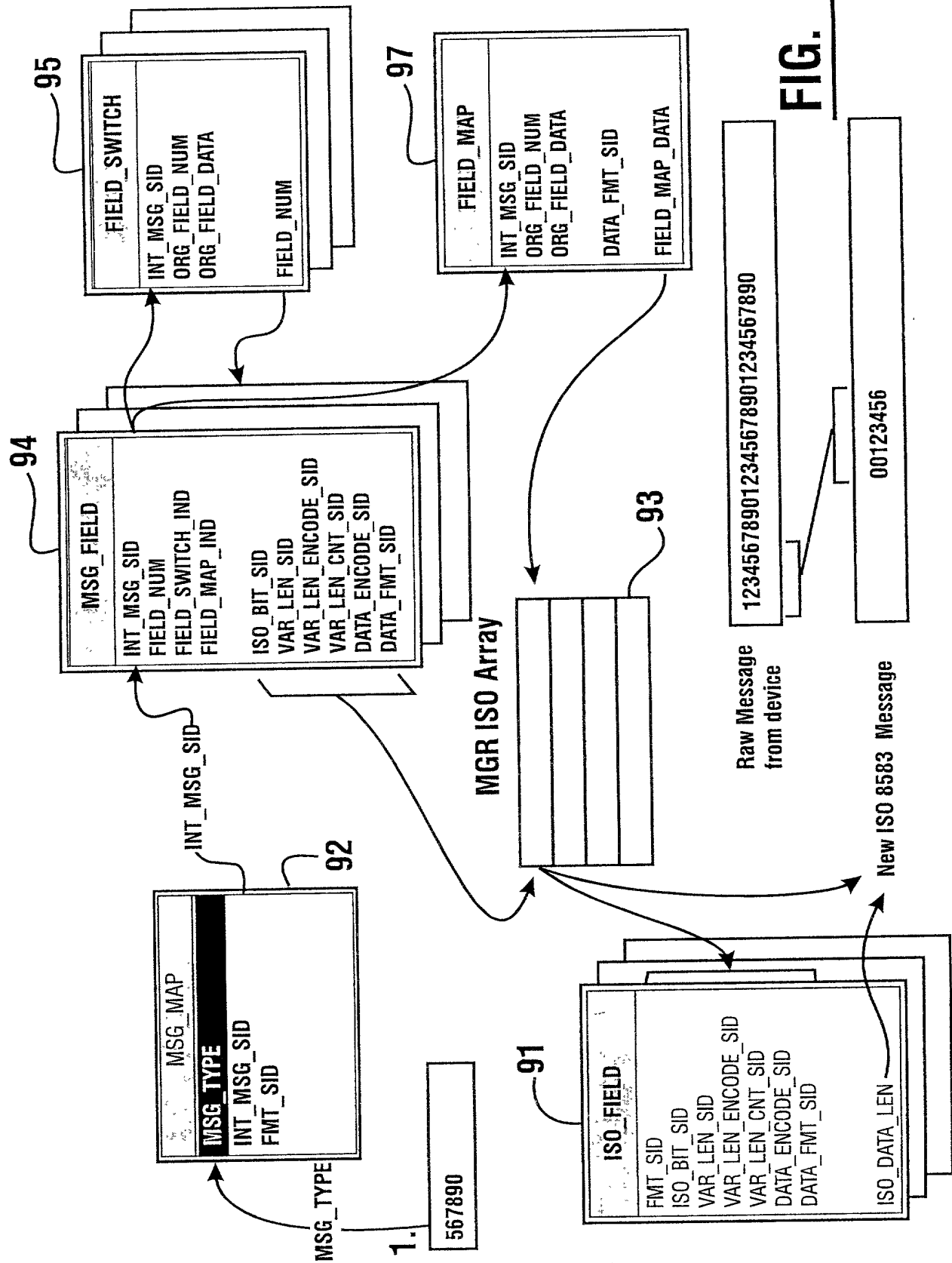


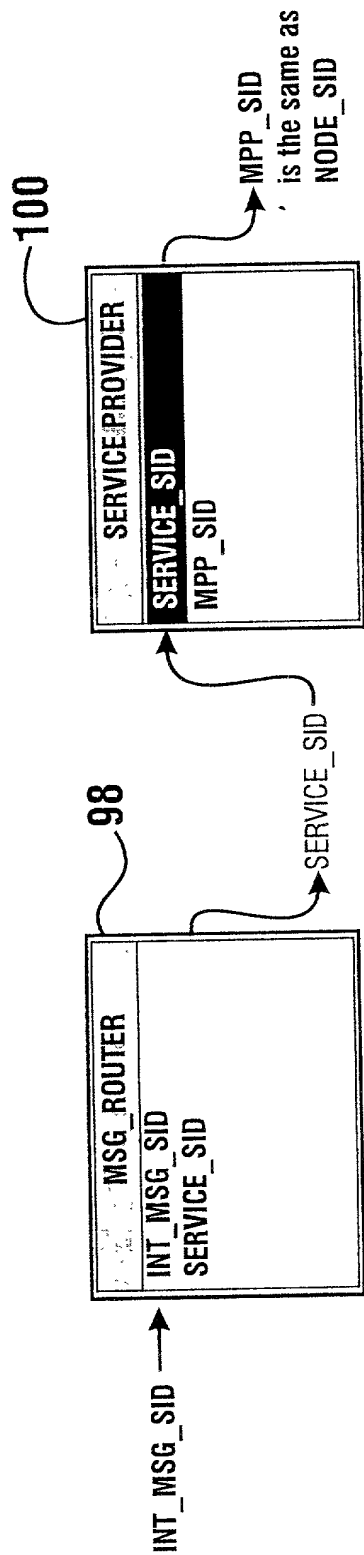
FIG. 10

1. The Listener buffers the information.
2. The MGR parses the ISO 8583 message into a message format that can be used by the device.
3. The Device driver sends the message to the device.
4. The POS terminal returns a message confirming the authorization message.

The message is then returned to the MPP in the same manner as before.







If the chosen Provider is not available at the time of the TCP/IP call. This table is used to determine if there is another service provider. Hot spare - fault tolerance.

FIG. 14

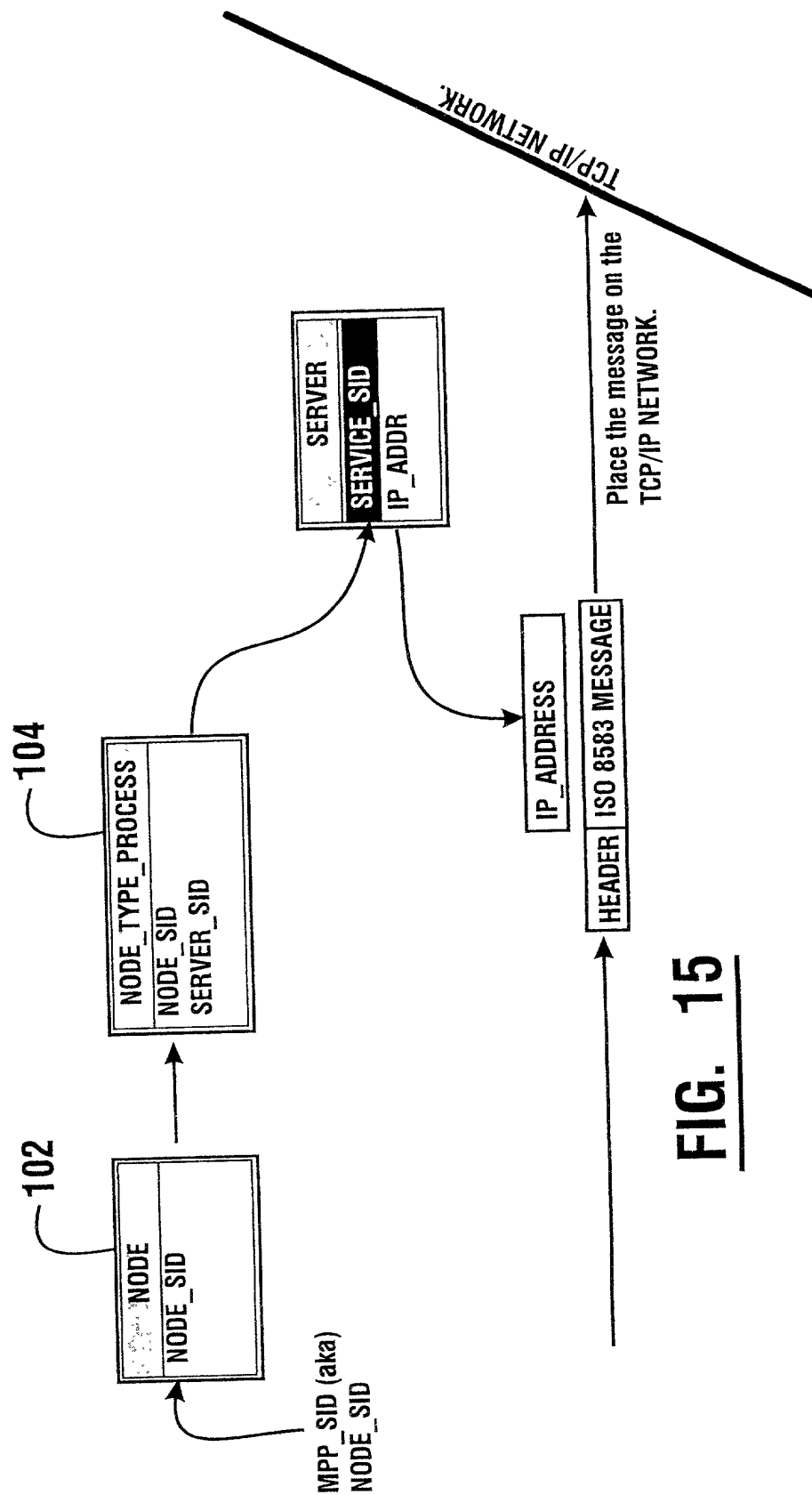


FIG. 15

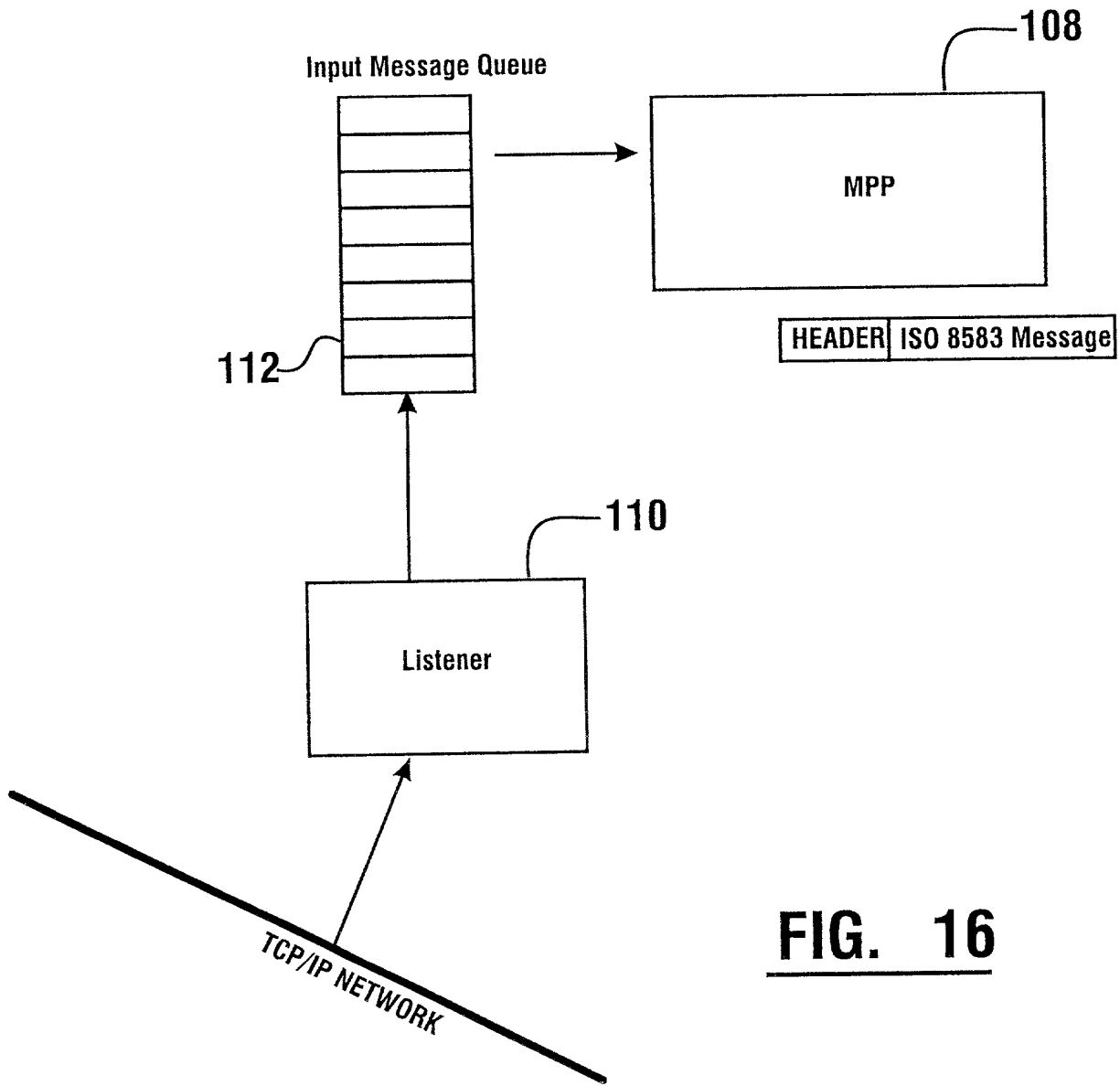


FIG. 16

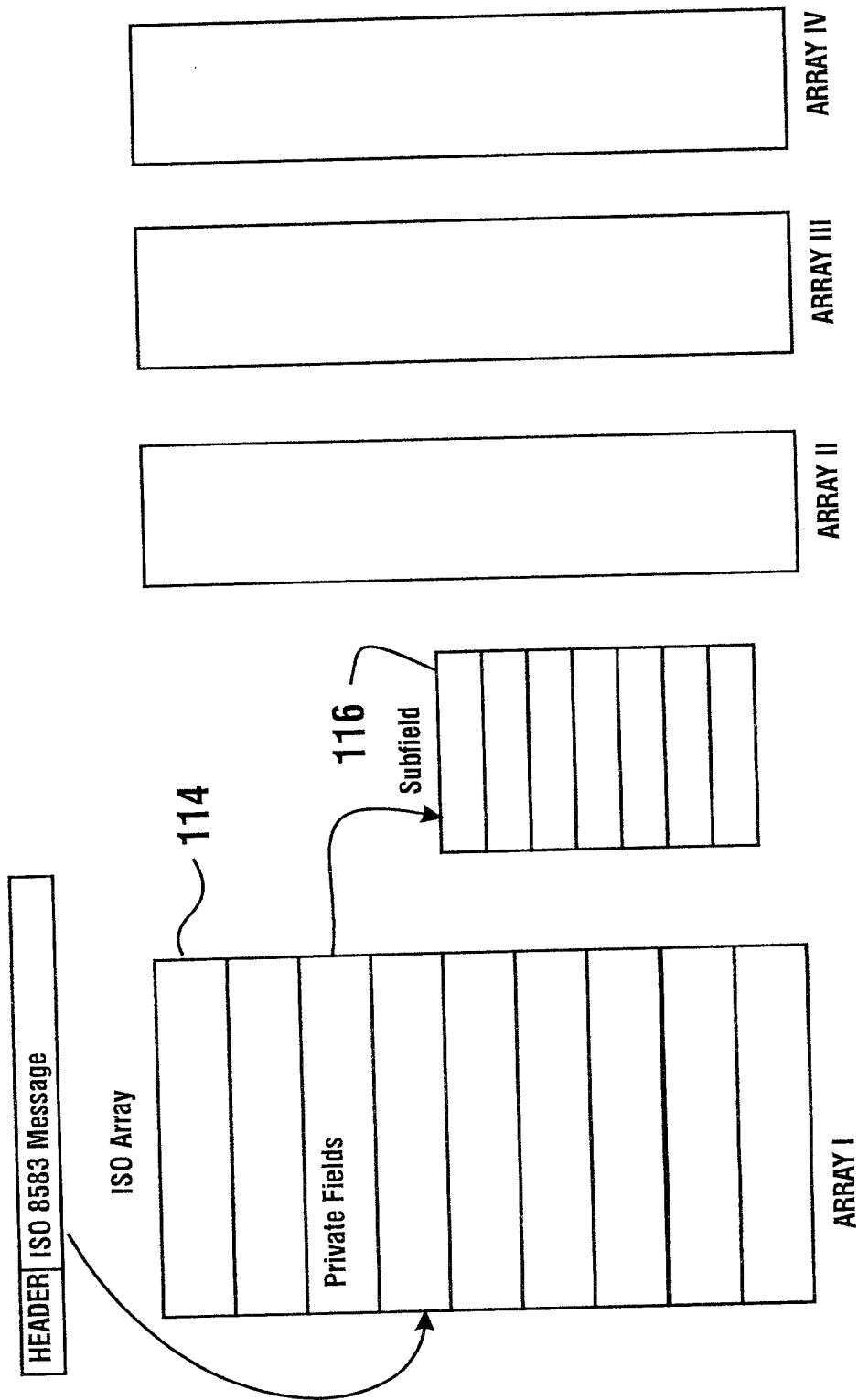


FIG. 17

FIG. 18

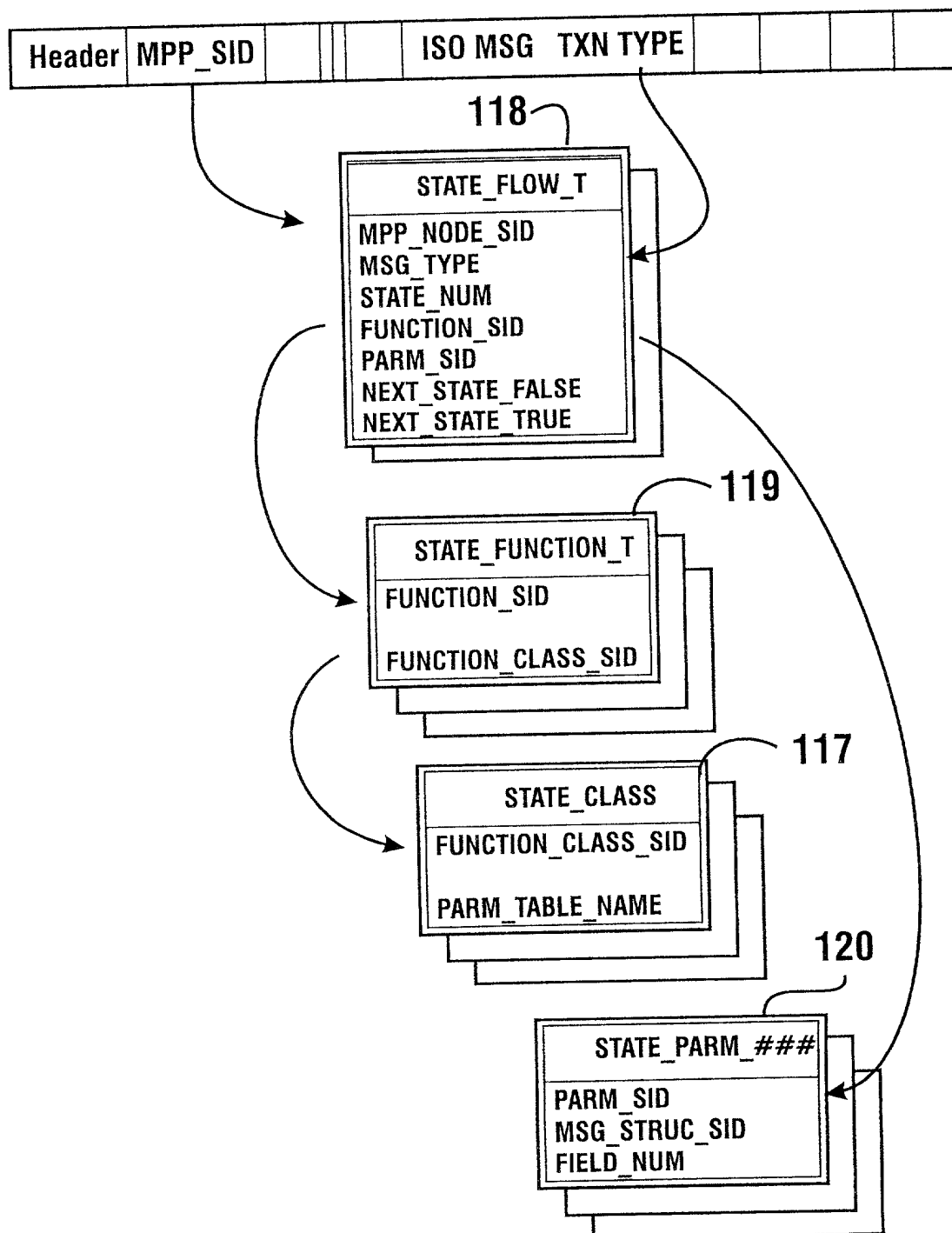


FIG. 18

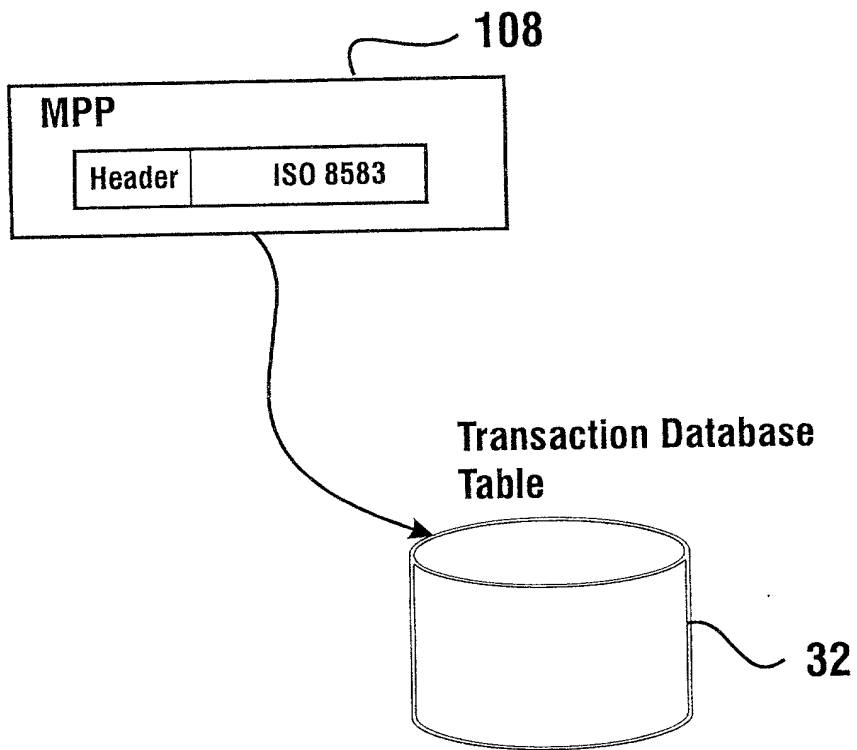


FIG. 19

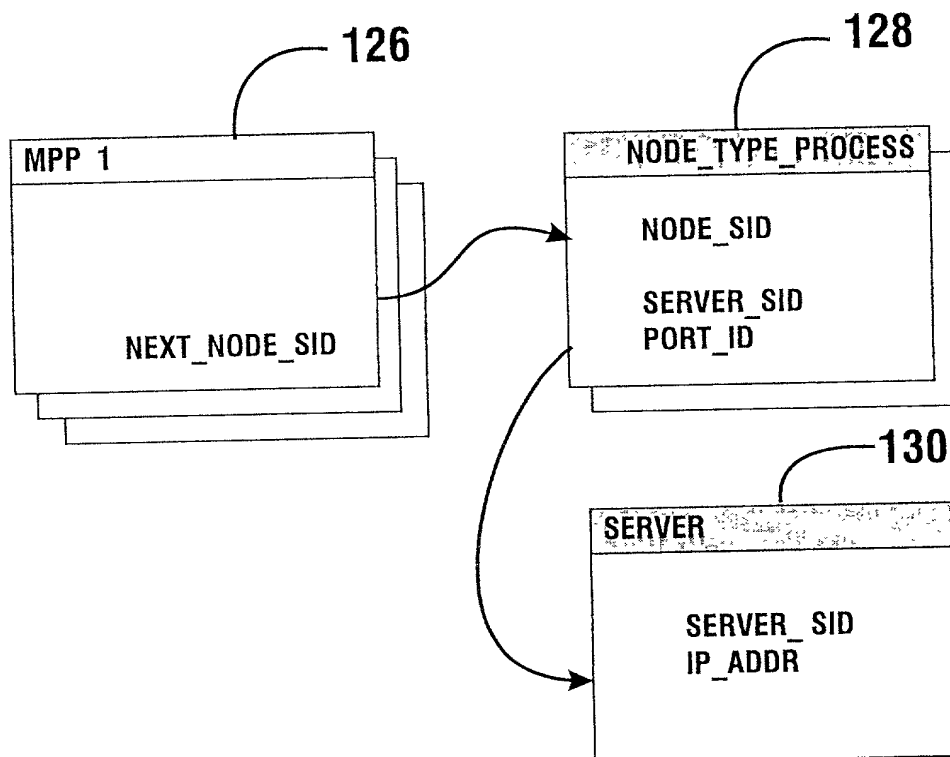


FIG. 20

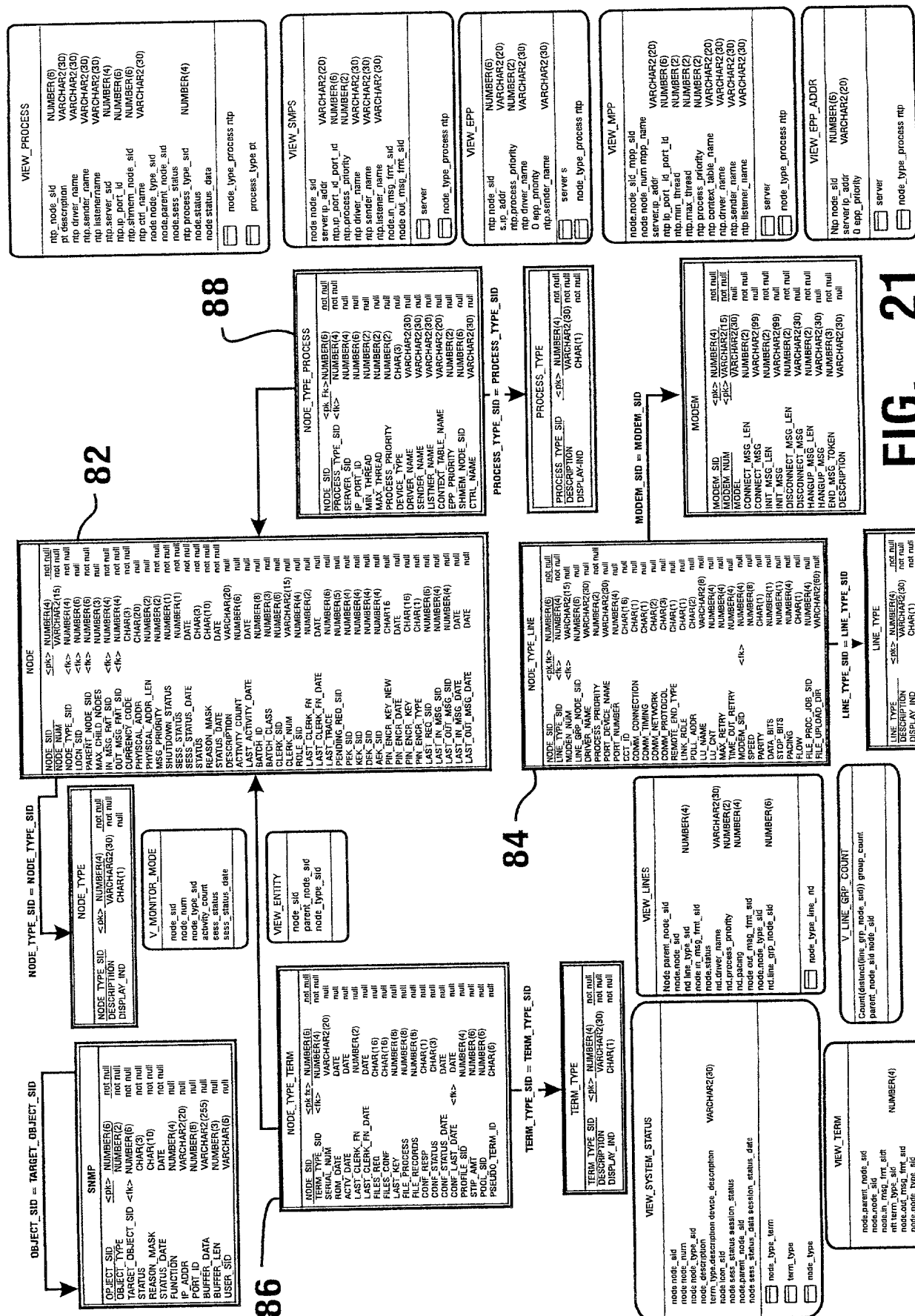


FIG. 21



FIG. 22

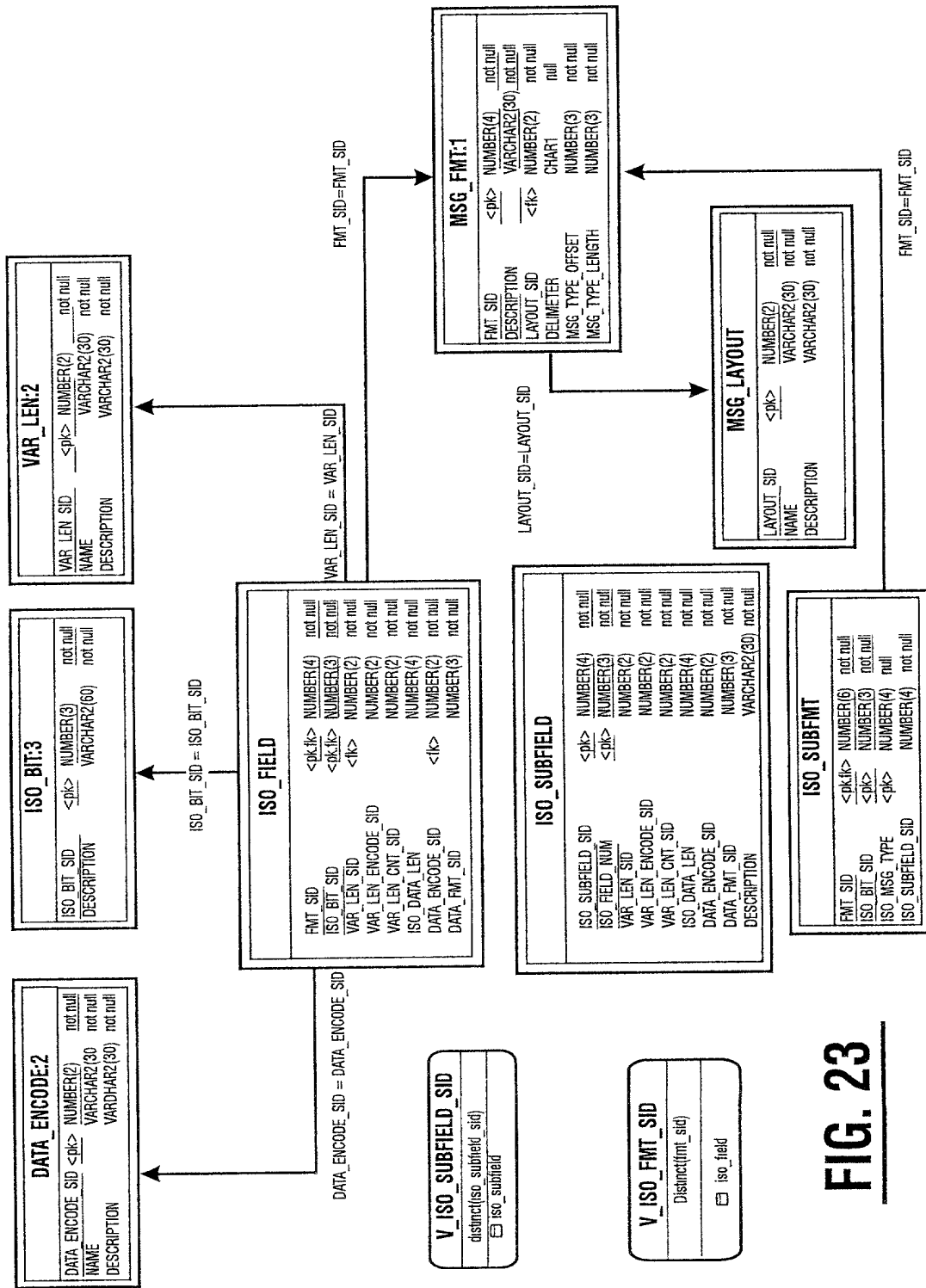


FIG. 23

EXTERNAL_HOST		
HOST_SID	<pk> NUMBER(6)	not null
HOST_NUM	VARCHAR2(15)	null
NAME	VARCHAR2(30)	null
ADDR	VARCHAR2(30)	null
CITY	VARCHAR2(20)	null
STATE	CHAR(2)	null
COUNTRY_CODE	CHAR(3)	null
ZIP_CODE	CHAR(9)	null
CONTACT_NAME	VARCHAR2(30)	null
TELEPHONE	VARCHAR2(16)	null
NODE_SID	NUMBER(6)	null
COMMENTS	VARCHAR2(30)	null
STATUS	CHAR(3)	null
STATUS_DATE	DATE	null

SERVER_HOST_LINK		
SERVER_SID	<pk fk> NUMBER(6)	not null
HOST_SID	<pk fk> NUMBER(6)	not null
PRIORITY	NUMBER(2)	null

SERVER_SID = SERVER_SID

SERVER		
SERVER_SID	<pk> NUMBER(6)	not null
NAME	VARCHAR2(20)	not null
IP_ADDR	VARCHAR2(20)	not null

COL_VALUE		
TABLE_NAME	<pk> VARCHAR2(20)	not null
COLUMN_NAME	<pk> VARCHAR2(20)	not null
ITEM_OFFSET	<pk> NUMBER(2)	not null
COLUMN_VALUE	<pk> VARCHAR2(3)	not null
DESCRIPTION	VARCHAR2(30)	not null

STATUS_REASON		
TABLE_NAME	<pk> VARCHAR2(20)	not null
STATUS_VALUE	<pk> CHAR(3)	not null
REASON_NUM	<pk> NUMBER(2)	not null

SYSTEM_PARM		
SYSTEM_PARM_SID	<pk> NUMBER(4)	not null
PARAMETER	<pk> VARCHAR2(10)	not null
VALUE	VARCHAR2(20)	not null
FMT	VARCHAR2(10)	not null
STATUS	CHAR(3)	not null
STATUS_DATE	DATE	not null
DESCRIPTION	VARCHAR2(30)	not null

FIG. 24

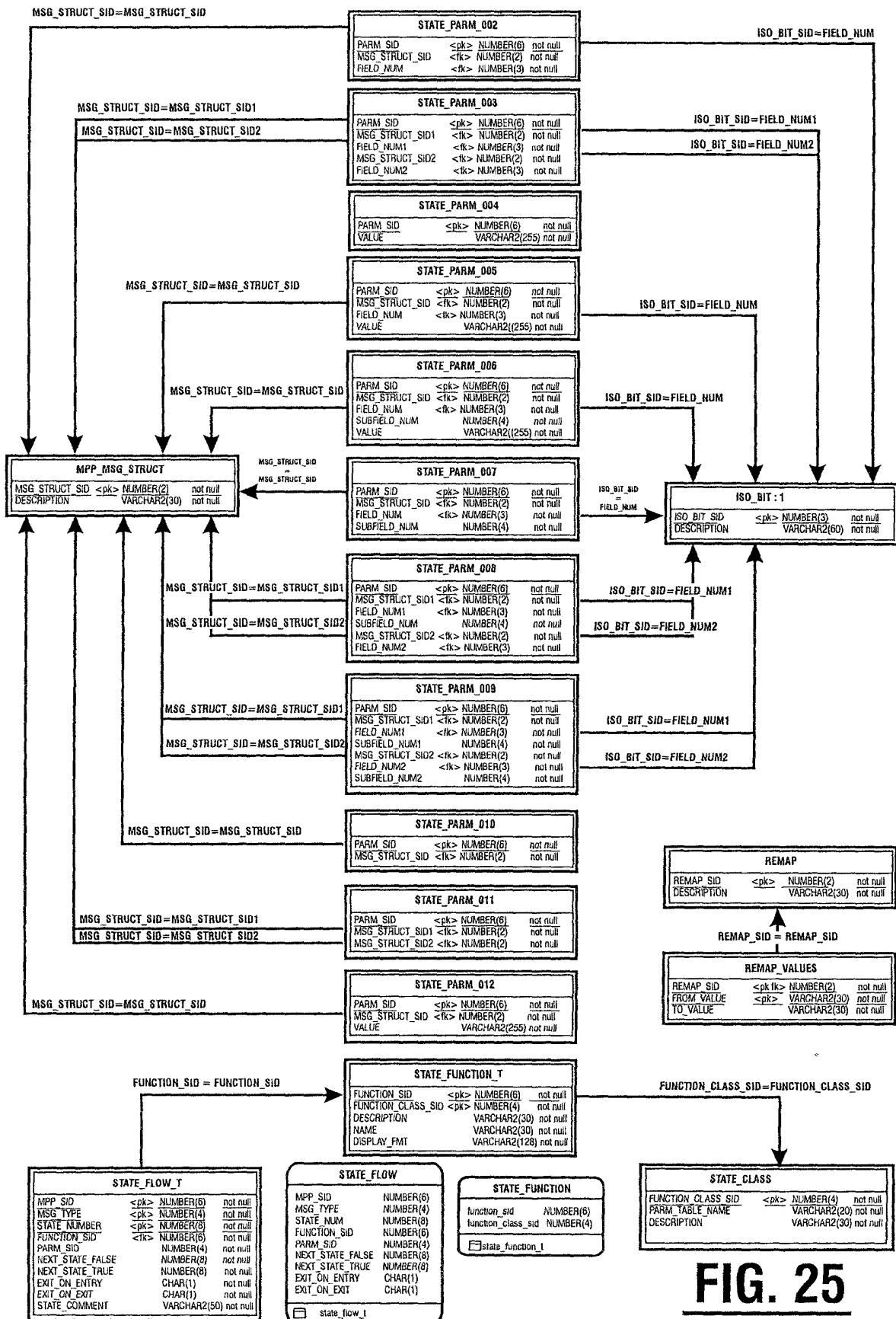


FIG. 25

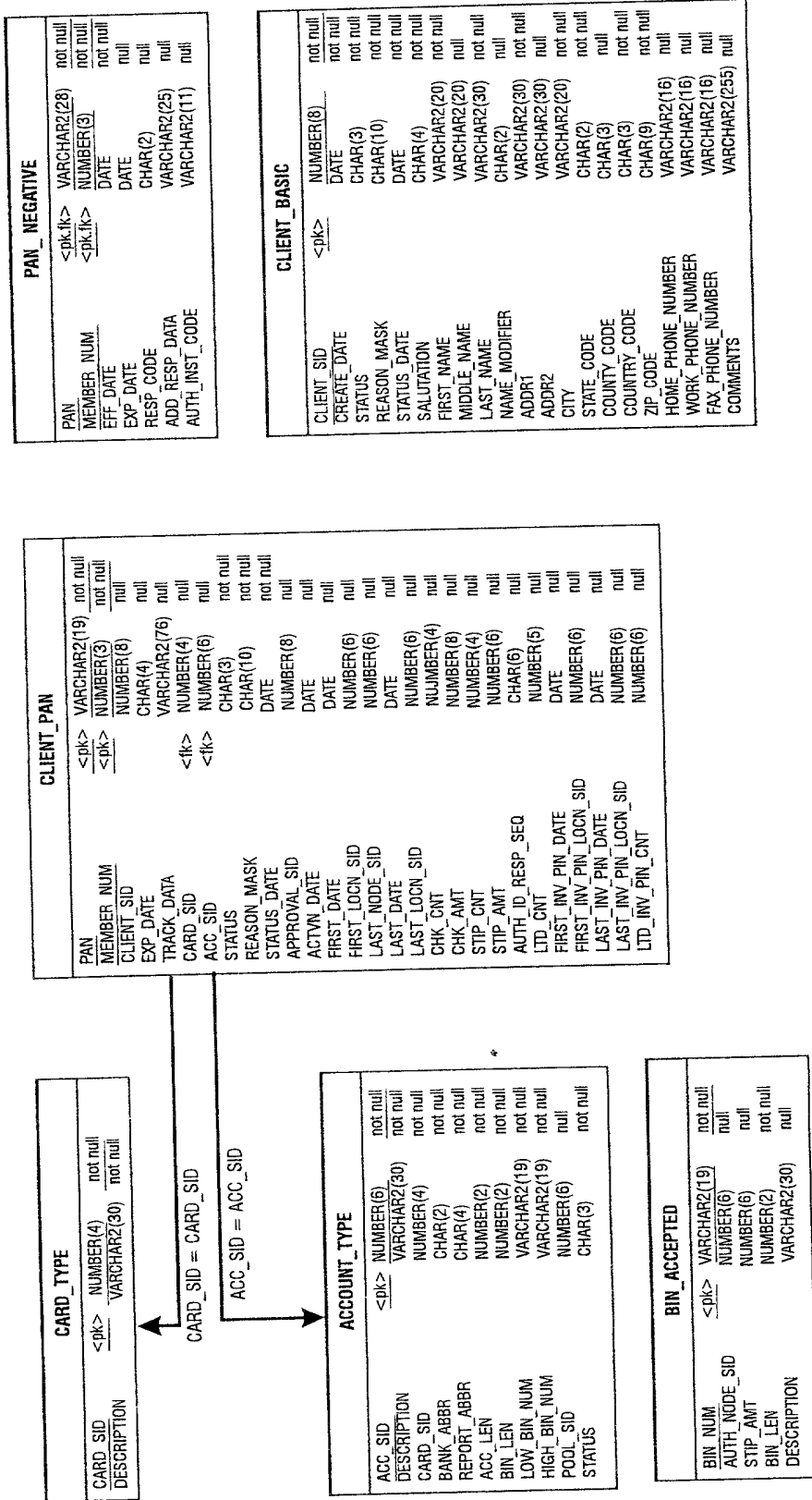


FIG. 26

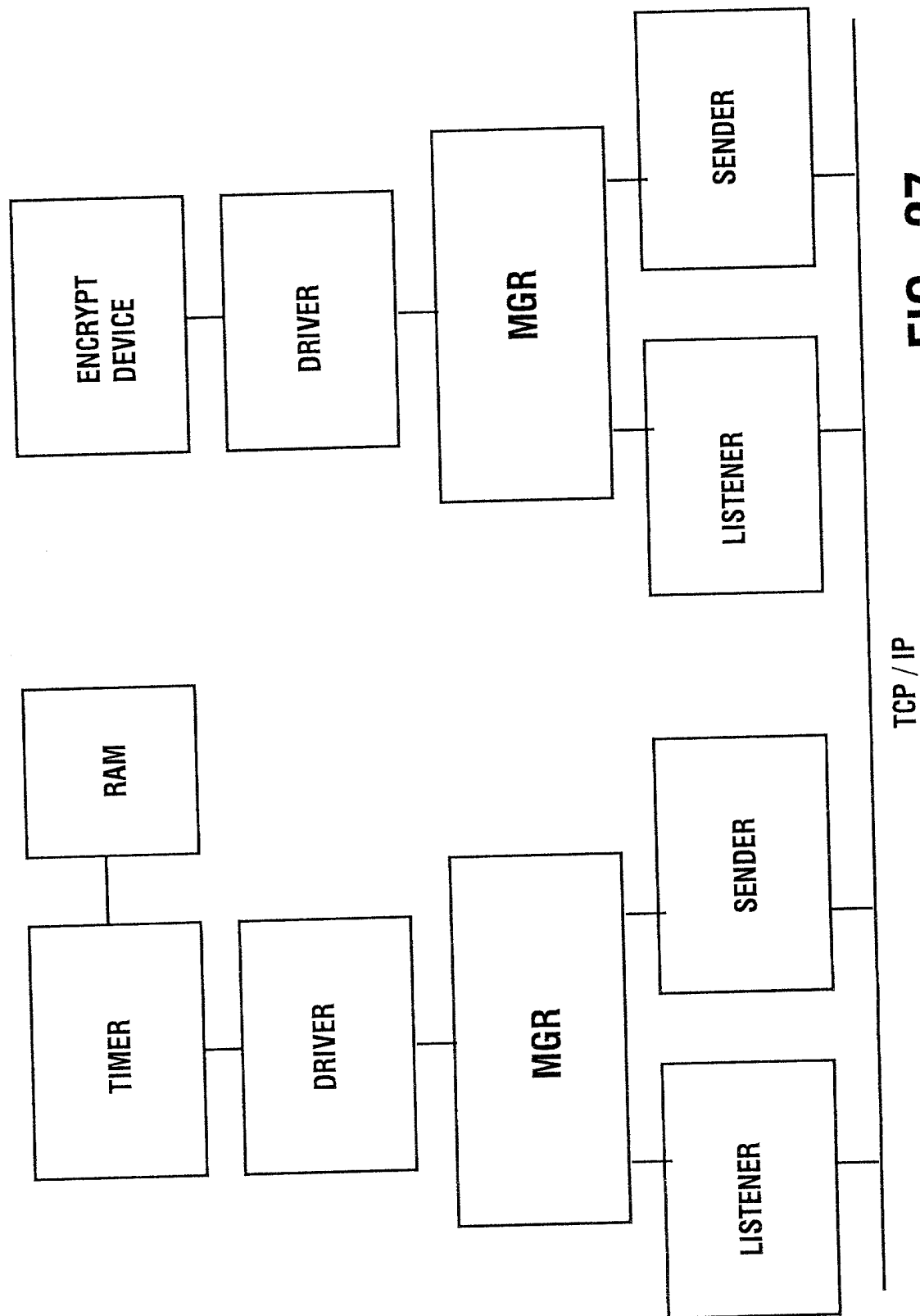


FIG. 27

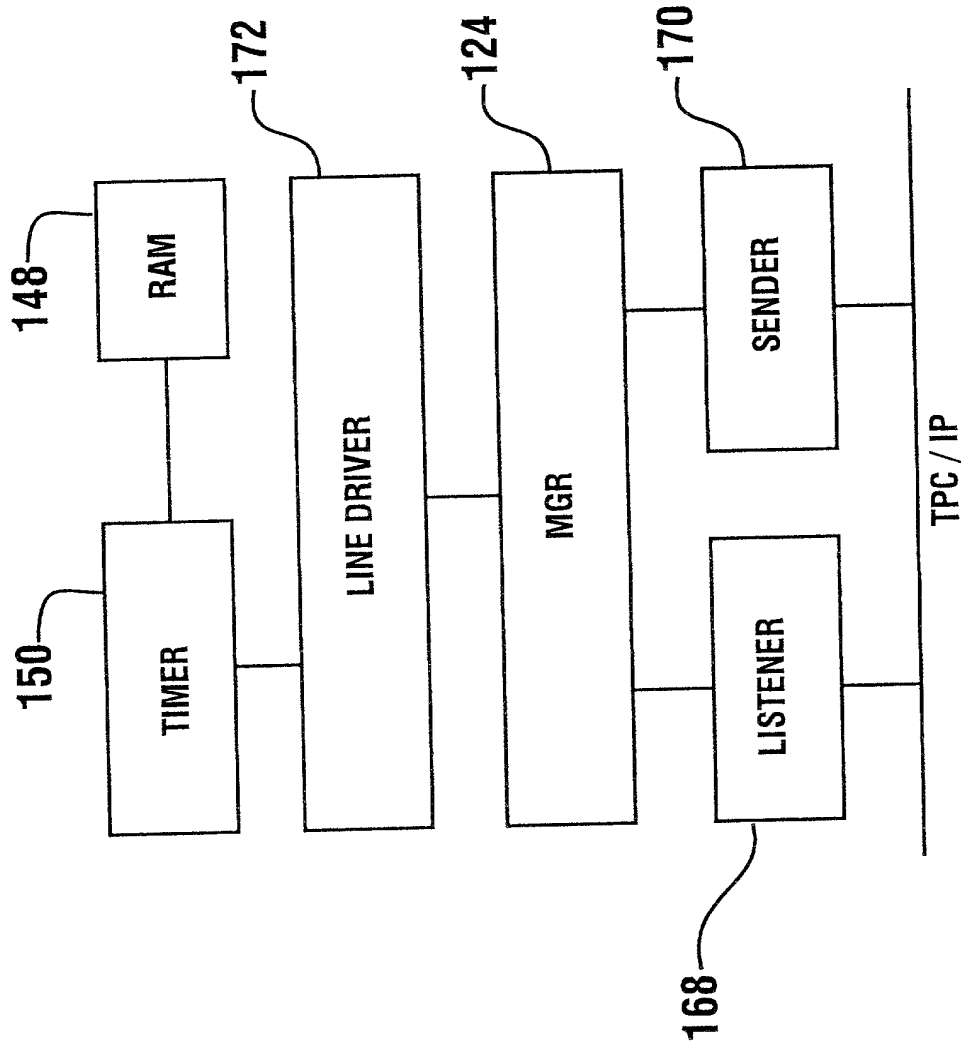


FIG. 28

MGS_ROUTER				
LINE_SID	<pk>	NUMBER(6)	not null	
NODE_SID	<pk>	NUMBER(6)	not null	
INT_MSG_SID	<pk>	NUMBER(6)	not null	
SERVICE_SID	<pk>	NUMBER(4)	not null	

SERVICE_SID = SERVICE_SID

SERVICE				
SERVICE_SID	<pk>	NUMBER(4)	not null	
DESCRIPTION		VARCHAR2(30)	not null	

SERVICE_SID = SERVICE_SID

SERVICE_PROVIDER				
SERVICE_SID	<pk>	NUMBER(4)	not null	
PATH_ORIGINAL	<pk>	NUMBER(2)	not null	
MPP_SID		NUMBER(6)	not null	
PRIORITY		NUMBER(2)	not null	

FIG. 29

FIG. 30

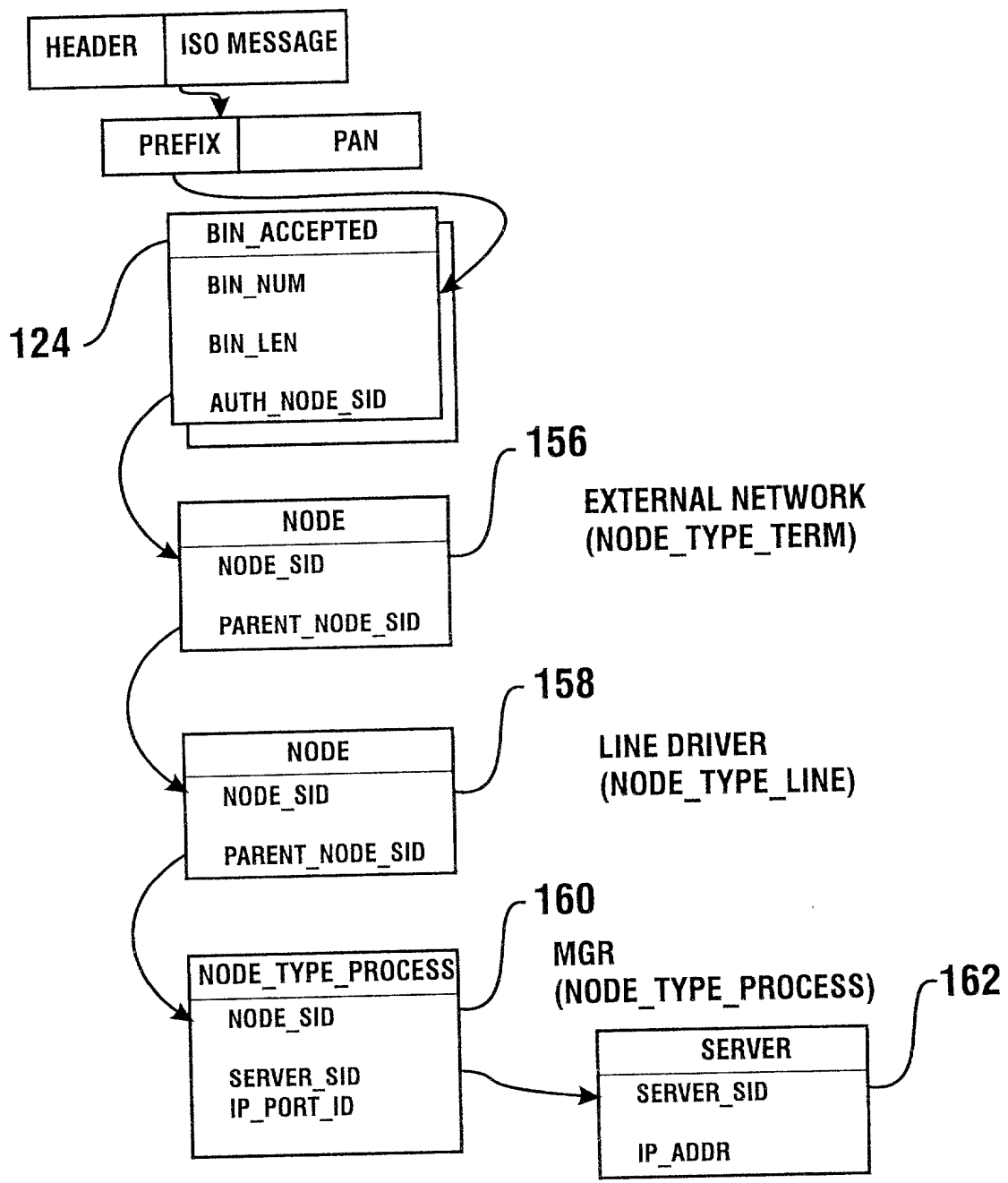


FIG. 30